Please cancel claims 1-14 and replace them with claims 15-32 as follows:

15. (New) A computer implemented replacement selection method comprising:

        creating a binary tree with a root node and multiple external nodes, wherein a quantity of nodes in said tree is equal to a number of input data streams to be processed through said tree, each node having a first integer identifying one of at least three input data streams and a status identifier identifying a status of each of said input data streams;

        processing a data item from each of said at least three input streams by placing one data item from each of said three input stream in each of said external nodes of said tree, respectively;

        promoting said data items of said nodes through said binary tree in a hierarchical manner, including comparing a first status identifier of a first node with a second status identifier of a second node, and if said status identifiers are unequal promoting one of said data items of one said nodes to a next hierarchical node in said tree, and switching a next comparison to a third status identifier of a third node with one of said other nodes in said tree;

        compiling a single output stream from said promoted data items, said output stream being a stream separate from said input streams, by copying compared data items promoted to said root node of said binary tree; and

        resolving comparison of duplicate identifiers through an identification process comprising:

                comparing a status identifier of two of said nodes;

                remembering at each node when a first data item in a fist of said nodes remained in said node and was a duplicate of a second data item in a second of said nodes, wherein said second data items was promoted to a next hierarchical node in said tree;

                promoting said data item from said first node to a next hierarchical node in said tree absent a comparison of said data items in said nodes when a next time a

said first node is designated for a key comparison.

16. (New)   The method of claim 15, wherein the step of promoting said data item from said first node to a next hierarchical node in said tree absent a comparison of said data items includes skipping a comparison of said remembered first data item with data items in other nodes of said tree.

17. (New)   The method of claim 15, wherein said status identifier has a value corresponding to empty, duplicate, merging, and done.

18. (New)   The method of claim 17, wherein said status identifier is an integer variable, and said status identifier value corresponding to empty is a value of zero, said status identifier value corresponding to duplicate is a value of one, said status identifier value corresponding to merging is a value of two, and a status identifier value corresponding to dine is a value of three.

19. (New)   The method of claim 15, wherein the replacement selection method is a loser-oriented selection tree,

20. (New)   The method of claim 15, wherein the step of promoting said data item from said first node to a next hierarchical node in said tree absent a comparison of said data items includes avoiding exhausting promoting duplicate data items from a single input data stream.

21. (New)   A system for processing computer-readable data comprising:
              a binary tree with a root node and multiple external nodes, wherein a quantity of nodes in said tree is equal to a number of input data streams to be processed through said tree, each node having a first integer to identify one of at least three input data streams and a status identifier to identify a status of each of said input data streams;

a data item to be processed from each of said at least three input streams by placement of one data item from each of said three input stream in each of said external nodes of said tree, respectively;

instructions to promote said data items of said nodes through said binary tree in a hierarchical manner, including comparison of a first status identifier of a first node with a second status identifier of a second node, and if status identifiers are unequal to promote one of said data items of one said nodes to a next hierarchical node in said tree, and to switch a next comparison to a third status identifier of a third node with one of said other nodes in said tree;

a single output stream compiled from said promoted data items, said output stream being a stream separate from said input streams, by copy of compared data items promoted to said root node of said binary tree; and

instructions to resolve comparison of duplicate identifiers through an identification process comprising:

comparison of a status identifier of two of said nodes;

remembering at each node when a first data item in a fist of said nodes remained in said node and was a duplicate of a second data item in a second of said nodes, wherein said second data items was promoted to a next hierarchical node in said tree;

promotion of said data item from said first node to a next hierarchical node in said tree absent a comparison of said data items in said nodes when a next time a said first node is designated for a key comparison.

22. (New)   The system of claim 21, wherein the instructions to promote said data item from said first node to a next hierarchical node in said tree absent a comparison of said data items includes skip of a comparison of said remembered first data item with data items in other nodes of said tree.

23. (New)   The system of claim 21, wherein said status identifier has a value corresponding

4

to empty, duplicate, merging, and done.

24. (New)   The system of claim 23, wherein said status identifier is an integer variable, and said status identifier value corresponding to empty is a value of zero, said status identifier value corresponding to duplicate is a value of one, said status identifier value corresponding to merging is a value of two, and a status identifier value corresponding to dine is a value of three.

25. (New)   The system of claim 21, wherein the promotion instruction is a loser-oriented selection tree.

26. (New)   The system of claim 21, wherein the instructions to promote said data item from said first node to a next hierarchical node in said tree absent a comparison of said data items includes instructions to avoid exhausting promotion of duplicate data items from a single input data stream.

27. (New)   An article comprising:
    a computer-readable data storage medium;
    instructions in the medium for implementing a replacement selection method comprising:
        instructions for creating a binary tree with a root node and multiple external nodes, wherein a quantity of nodes in said tree is equal to a number of input data streams to be processed through said tree, each node having a first integer identifying one of at least three input data streams and a status identifier identifying a status of each of said input data streams;
        instructions for processing a data item from each of said at least three input streams by placing one data item from each of said three input stream in each of said external nodes of said tree, respectively;
        instructions for promoting said data items of said nodes through said binary tree in a hierarchical manner, including comparing a first status identifier of a first node with a

5

second status identifier of a second node, and if said status identifiers are unequal promoting one of said data items of one said nodes to a next hierarchical node in said tree, and switching a next comparison to a third status identifier of a third node with one of said other nodes in said tree;

      instructions for compiling a single output stream from said promoted data items, said output stream being a stream separate from said input streams, by copying compared data items promoted to said root node of said binary tree; and

      instructions for resolving comparison of duplicate identifiers through an identification process comprising:

            comparing a status identifier of two of said nodes;

            remembering at each node when a first data item in a fist of said nodes remained in said node and was a duplicate of a second data item in a second of said nodes, wherein said second data items was promoted to a next hierarchical node in said tree;

            promoting said data item from said first node to a next hierarchical node in said tree absent a comparison of said data items in said nodes when a next time a said first node is designated for a key comparison.

28. (New)   The article of claim 27, wherein the instructions for promoting said data item from said first node to a next hierarchical node in said tree absent a comparison of said data items includes skipping a comparison of said remembered first data item with data items in other nodes of said tree.

29. (New)   The article of claim 27, wherein said status identifier has a value corresponding to empty, duplicate, merging, and done.

30. (New)   The article of claim 29, wherein said status identifier is an integer variable, and said status identifier value corresponding to empty is a value of zero, said status identifier value corresponding to duplicate is a value of one, said status identifier value corresponding to merging is a value of two, and a status identifier value

corresponding to dine is a value of three.

31. (New)   The article of claim 27, wherein the replacement selection method is a loser-oriented selection tree,

32. (New)   The article of claim 27, wherein the instructions for promoting said data item from said first node to a next hierarchical node in said tree absent a comparison of said data items includes avoiding exhausting promoting duplicate data items from a single input data stream.